

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-312151

(43)Date of publication of application : 09.11.1999

(51)Int.Cl.

G06F 15/16

G06F 15/16

G06F 9/46

G06F 13/00

(21)Application number : 10-118121

(71)Applicant : HITACHI LTD

(22)Date of filing : 28.04.1998

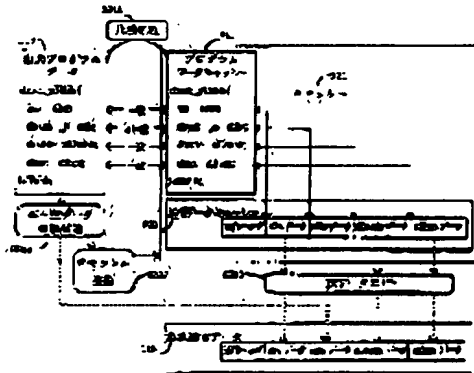
(72)Inventor : IKOMA MIKIO  
FUJIWARA YOJI  
TANIDA KOUSUKE

## (54) HIGH-SPEED DISTRIBUTION OBJECT REQUEST BROKER

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To obtain a high-speed processing system by providing a marshalling part for retrieving whether a data form reliant on a program language is stored in a cache or not so as to execute conversion through the use of data on the cache or to convert the data form into data, which is made into a stream.

**SOLUTION:** A marshalling part compares program data to be transmitted 113 with program data 211 on the corresponding cache at first and checks as to whether or not the contents is the same. When the contents are the same, succeeding data are checked. When there is difference between them, communication data on the cache corresponding to program data which are same until the difference is detected are copied on request communication data. It is checked whether program data with difference are a basic type or a user defining type, a conversion processing is executed in communication data when it is basic type and a marshalling processing is recursively called when it is the user-defined type. A conversion result is registered in the cache.



## LEGAL STATUS

[Date of request for examination] 19.09.2001

[Date of sending the examiner's decision of rejection] 04.10.2005

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

Searching PAJ

2/2 ページ

[Date of registration]

[Number of appeal against examiner's decision  
of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-312151

(43) 公開日 平成11年(1999)11月9日

(51) Int. Cl. <sup>7</sup>	識別記号	F I	
G 0 6 F 15/16	4 3 0	G 0 6 F 15/16	4 3 0 B
	3 7 0		3 7 0 N
9/46	3 6 0	9/46	3 6 0 B
			3 6 0 F
13/00	3 5 1	13/00	3 5 1 F
審査請求 未請求 請求項の数 2 O L (全 6 頁)			

(21) 出願番号 特願平10-118121

(22) 出願日 平成10年(1998)4月28日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 居駒 幹夫

神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア開発本部内

(72) 発明者 藤原 要治

神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア開発本部内

(72) 発明者 谷田 耕教

神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア開発本部内

(74) 代理人 弁理士 小川 勝男

(54) 【発明の名称】 高速な分散オブジェクトリクエストブローカ

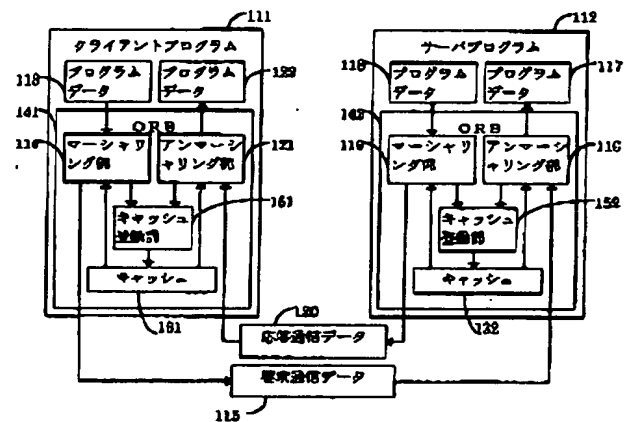
(57) 【要約】

【課題】分散オブジェクト間の要求や応答を処理するオブジェクトリクエストブローカ（以下、ORB）において、ORB間の通信で使用する特定の計算機に依存しないデータ（通信データ）から特定のプログラム言語依存のデータ形式（プログラムデータ）への変換処理を高速化する。

【解決手段】ORBが送受信した通信データおよび、それに対応するプログラムデータへの変換の対応を高速アクセスが可能な記憶装置にキャッシュとして格納する。2回目以降にキャッシュされた同一の通信データの受信又はプログラムデータの送信処理が発生した場合にはキャッシュされている変換結果を使用する。

【効果】

(図1)



(2)

特開平11-312151

**【特許請求の範囲】**

**【請求項1】** 分散コンピューティング環境において特定の計算機上の特定のプログラム言語依存のデータ形式を特定の計算機に依存しないストリーム化されたデータに変換する装置において、

プログラム言語依存のデータ形式とストリーム化されたデータの対応を記憶するキャッシュ部と、

プログラム言語依存のデータ形式の全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しない場合は、キャッシュを使用せずにプログラム言語依存のデータ形式をストリーム化されたデータに変換するマーシャリング部と、

プログラム言語依存のデータ形式をストリーム化されたデータに変換、または、ストリーム化されたデータをプログラム言語依存のデータ形式に変換時に、その結果をキャッシュに登録するキャッシュ登録部を具えることを特徴とする装置。

**【請求項2】** 分散コンピューティング環境において特定の計算機に依存しないストリーム化されたデータを特定の計算機上の特定のプログラム言語依存のデータ形式に変換する装置において、

ストリーム化されたデータとプログラム言語依存のデータ形式の対応を記憶するキャッシュ部と、

ストリーム化されたデータの全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しなかった場合は、キャッシュを使用せずにストリーム化されたデータをプログラム言語依存のデータ形式に変換するアンマーシャリング部と、

ストリーム化されたデータをプログラム言語依存のデータ形式に変換、または、プログラム言語依存のデータ形式をストリーム化されたデータに変換時に、その結果をキャッシュに登録するキャッシュ登録部を具えることを特徴とする装置。

**【発明の詳細な説明】****【0001】**

**【発明の属する技術分野】** 本発明は分散コンピューティングを行う複数計算機上の分散オブジェクト間の通信に使用され、計算機内の特定のプログラム言語のデータ形式と計算機間の通信で用いられるストリーム型のデータとの相互変換方法に関するものである。

**【0002】**

**【従来の技術】** 一般的にあるプログラムから他のプログラムを呼び出すときは、同一計算機上で特定のオペレーティングシステム、特定のプログラム言語に依存したインタフェースを使用してデータの送受信を行っている。これに対して、分散コンピューティング環境においては、複数のプログラム（分散オブジェクト）が複数の計算機に跨って存在でき送受信できる。ここで、処理を要

求する分散オブジェクトをクライアントオブジェクトと呼び、クライアントオブジェクトからの要求を処理する分散オブジェクトをサーバオブジェクトと呼ぶ。また、クライアントオブジェクトからの要求を受け付け、要求を処理する適切なサーバオブジェクトを検索し、サーバオブジェクトの存在する計算機と通信を行い、サーバオブジェクトを呼び出す機能をオブジェクトリクエストブローカ（ORB）と呼ぶ。複数計算機間でクライアントオブジェクトとサーバオブジェクトが通信する場合は、ORBが特定の計算機のアーキテクチャ、特定のオペレーティングシステム、特定のプログラム言語に依存しない中立的なストリーム形式のデータを用いてデータの交換を行う。従って、ORBは特定の計算機のアーキテクチャ、特定のオペレーティングシステム、特定のプログラム言語形式から中立的なデータストリームに変換する処理（マーシャリング）及び、その逆の処理（アンマーシャリング）が必要になる。

**【0003】**

**【発明が解決しようとする課題】** 一般にサーバオブジェクトはある計算機上に常駐的に存在し、不特定多数のクライアントオブジェクトからの要求を複数回処理する。クライアントオブジェクトからの要求が来るたびにアンマーシャリング処理が発生し、その要求に対する応答を返すごとにマーシャリング処理が発生する。複雑なデータ形式がクライアントとサーバ間で交換される場合、そのマーシャリング、アンマーシャリングに要する計算機資源（CPU、メモリ容量）の負荷は大きい。本発明では、複数回クライアント、サーバ間で送受信が発生した場合の高速な処理方式を提供することを目的とする。

**【0004】**

**【課題を解決するための手段】** 上記の目的を達成するため、特定の計算機上の特定のプログラム言語依存のデータ形式を特定の計算機に依存しないストリーム化されたデータに変換する装置において、プログラム言語依存のデータ形式とストリーム化されたデータの対応を記憶するキャッシュ部と、プログラム言語依存のデータ形式の全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しない場合は、キャッシュを使用せずにプログラム言語依存のデータ形式をストリーム化されたデータに変換するマーシャリング部と、プログラム言語依存のデータ形式をストリーム化されたデータに変換、または、ストリーム化されたデータをプログラム言語依存のデータ形式に変換時に、その結果をキャッシュに登録するキャッシュ登録部を具える。このような構成で、データを送信時、マーシャリング部は受信したデータがキャッシュ上に存在しているかどうかを検索し、データが存在する場合はキャッシュを参照して高速に変換し、存在していなかった場合はキャッシュを用いずに通常の方法で低速にデータを変換す

(3)

特開平11-312151

る。通常の方法で変換したときは変換結果をキャッシュ登録部を使用してキャッシュに登録する。

【0005】本発明では、特定の計算機に依存しないストリーム化されたデータを特定の計算機上の特定のプログラム言語依存のデータ形式に変換する装置において、ストリーム化されたデータとプログラム言語依存のデータ形式の対応を記憶するキャッシュ部と、ストリーム化されたデータの全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しなかった場合は、キャッシュを使用せずにストリーム化されたデータをプログラム言語依存のデータ形式に変換するアンマーシャリング部と、ストリーム化されたデータをプログラム言語依存のデータ形式に変換、または、プログラム言語依存のデータ形式をストリーム化されたデータに変換時に、その結果をキャッシュに登録するキャッシュ登録部を具える。このような構成で、データを受信時、アンマーシャリング部は受信したデータがキャッシュ上に存在しているかどうかを検索し、データが存在する場合はキャッシュを用いて高速に変換し、存在していなかった場合はキャッシュを用いずに通常の方法で低速にデータを変換する。通常の方法で変換したときは変換結果をキャッシュ登録部を使用してキャッシュに登録する。

【0006】

【発明の実施の形態】図1は本発明の実施する分散コンピューティング環境の構成と本発明でのデータの流れを示した図である。図において、クライアントプログラム111内部には計算機や記述されたプログラムに依存したプログラムデータ113、122、リンケージされたプログラムORB141を内蔵している。また、サーバプログラム112内部には計算機や記述されたプログラムに依存したプログラムデータ117、118、リンケージされたプログラムORB142を内蔵している。

【0007】次に図1でどのようにクライアントプログラム111とサーバプログラム112がデータを交換するかをデータの流れで説明する。クライアントプログラム111はクライアントプログラムが動作する計算機や記述されたプログラムに依存したプログラムデータ113をサーバプログラム112に送信するため、ORB141のマーシャリング部114でプログラムデータ113から特定計算機やプログラム言語に非依存な通信データ115へ変換する。この時に、クライアントプログラムのキャッシュ131を参照し、プログラムデータ113の全て又は一部がキャッシュ131に存在すればそれを利用して要求通信データ115に変換する。変換後サーバプログラム112に送信する。

【0008】サーバプログラム112ではORB142のアンマーシャリング部116で受信した要求通信データ115をサーバプログラムが動作する計算機や記述さ

れたプログラムに依存したプログラムデータ117に変換する。この時に、サーバプログラムのキャッシュ132を参照し、要求通信データ115の全て又は一部がキャッシュ132に存在すればそれを利用してプログラムデータ117に変換する。

【0009】サーバプログラムはクライアントからの要求を処理後、結果のプログラムデータ118をクライアントプログラムに返すため、ORB142のマーシャリング部119で応答通信データ120に変換する。この時に、サーバプログラムのキャッシュ132を参照し、プログラムデータ118の全て又は一部がキャッシュ132に存在すればそれを利用して応答通信データ120に変換する。変換された応答通信データ120はクライアントプログラムに送信される。

【0010】クライアントプログラム111では受信したデータをORB141のアンマーシャリング部121でクライアントプログラム依存のプログラムデータ122に変換する。この時に、クライアントプログラムのキャッシュ131を参照し、応答通信データ120の全て又は一部がキャッシュ131に存在すればそれを利用してプログラムデータ122に変換する。このようにして、分散したクライアントとサーバのプログラム間で高速なデータの受け渡しができる。

【0011】次に、請求項1に対応する図1のマーシャリング部114、119での処理の実施例を示す。本説明では図1のマーシャリング部114を例に採って説明するがマーシャリング部119でも同様な実施例が可能である。

【0012】図2はマーシャリング時のキャッシュの構造とマーシャリング処理のデータの流れを示した図である。まず、キャッシュの構造を説明する。キャッシュはプログラムデータの型毎にプログラムデータの内容とそれに対応する通信データの内容の対を格納している（図2の例ではaTable構造体に対するプログラムデータキャッシュ211と通信データキャッシュ212の対）。プログラムデータがユーザ定義の場合、キャッシュではユーザ定義型のプログラムデータの各要素（ユーザ定義型又は基本型）毎に、それに対応する通信データの先頭からのオフセットを記憶している。図2の例では、ユーザ定義型及び基本型毎にキャッシュを持つ方式を示している。本図ではキャッシュ上に一対の通信データとプログラムデータの対応を示したが、キャッシュ上には複数対の通信データとプログラムデータの対応を持つことも可能である。

【0013】次に、図2と図3を用い、マーシャリング処理のデータの流れと制御の流れを説明する。マーシャリング部ではまず、送信したいプログラムデータ113とその型に対応したキャッシュ上のプログラムデータ211を比較し（S311）、内容が同一か否かをチェックする（S312）。ここで、チェックするデータが構

(4)

特開平11-312151

造体や共用体等のユーザ定義型の場合も、そのユーザ定義型を構成する型へ分解せずに、構造体同士をメモリ上で比較する。その結果、内容が同一の場合は次のデータをチェックする。内容に差異があった場合は、差異を検出するまで同一だったプログラムデータに対応するキャッシュ上の通信データ212を要求通信データ115にコピーする(S313)。差異のあったプログラムデータが基本型かユーザ定義型かをチェックし(S314)基本型の場合はプログラムデータを通信データへ変換処理を行い(S315)、ユーザ定義型の場合はリカーシブにマーシャリング処理を呼び出す(S316)(図2の例では出力プログラムデータ113の不一致だった構造体aStrを引数にマーシャリング処理をリカーシブコールする)。S315またはS316での変換結果をキャッシュに登録する(S317)。S311からS317までの処理を全てのデータに対して実行し、データがなくなった場合(S310)その時点でキャッシュに残っている通信データをコピーする(S318)。

【0014】次に、図1のアンマーシャリング部116、121での処理の実施例を示す。本説明では図1のアンマーシャリング部116を例に採って説明するがアンマーシャリング部121でも同様な実施例が可能である。

【0015】図4はアンマーシャリング時のキャッシュの構造とアンマーシャリング処理のデータの流れを示した図である。まず、キャッシュの構造を説明する。キャッシュはプログラムデータの型毎にプログラムデータの内容とそれに対応する通信データの内容の対を格納している(図4の例ではaTable構造体に対する通信データキャッシュ411とプログラムデータキャッシュ412の対)。プログラムデータがユーザ定義型の場合、キャッシュではユーザ定義型のプログラムデータの各要素(ユーザ定義型又は基本型)毎に、それに対応する通信データの先頭からのオフセットを記憶している。この関連を利用して、通信データの任意のオフセットがプログラムデータのどの要素に対応しているかが分かる。図4の例では、ユーザ定義型及び基本型毎にキャッシュを持つ方式を示している。本図ではキャッシュ上に型毎に一对の通信データとプログラムデータの対応を示したが、キャッシュ上には複数対の通信データとプログラムデータの対応を持つことも可能である。

【0016】次に、図4と図5を用い、アンマーシャリング処理のデータの流れと制御の流れを説明する。アンマーシャリング部ではまず、受信した要求通信データ115とキャッシュ上の通信データ411をオクテット単位で比較し(S511)、同一の値を持っているか否かをチェックする(S512)。その結果、同一の値を持っている場合は次のオクテットをチェックする。同一の値を持っておらず差異があった場合は、差異を検出する

まで同一だった通信データに対応するキャッシュ上のプログラムデータ412を出力プログラムデータ117にコピーする(S513)。差異のあったプログラムデータが基本型かユーザ定義型かをチェックし(S514)基本型の場合は通信データをプログラムデータへ変換処理を行い(S515)、ユーザ定義型の場合はリカーシブにアンマーシャリング処理を呼び出す(S516)。S515またはS516での変換結果をキャッシュに登録する(S517)。S511からS517までの処理を通信データの全オクテットに対して実行し、データがなくなった場合(S510)その時点でキャッシュ上に残っているプログラムデータをコピーする(S518)。

#### 【0017】

【発明の効果】一般にサーバオブジェクトはある計算機上に常駐的に存在し、不特定多数のクライアントオブジェクトから同様な要求を複数回処理する。従来はこの複数回の同様な処理要求に対してマーシャリング、アンマーシャリング処理を独立に実行するため、複雑なデータ形式がクライアントとサーバ間で交換される場合、そのマーシャリング、アンマーシャリングに要する計算機資源(CPU、メモリ容量)の負荷は大きい。本発明では、一度処理した変換処理をキャッシュに格納して次回以降これを活用することによりマーシャリング、アンマーシャリングに要する計算機資源を低減することが可能となる。

#### 【図面の簡単な説明】

【図1】本発明のORBでの実施例の概要を示した図である。

【図2】本発明のマーシャリングの実施例でのデータの流れとキャッシュの構造を示す図である。

【図3】本発明のマーシャリングの実施例での処理の流れを示すフローチャートである。

【図4】本発明のアンマーシャリングの実施例でのデータの流れとキャッシュの構造を示す図である。

【図5】本発明のアンマーシャリングの実施例での処理の流れを示すフローチャートである。

#### 【符号の説明】

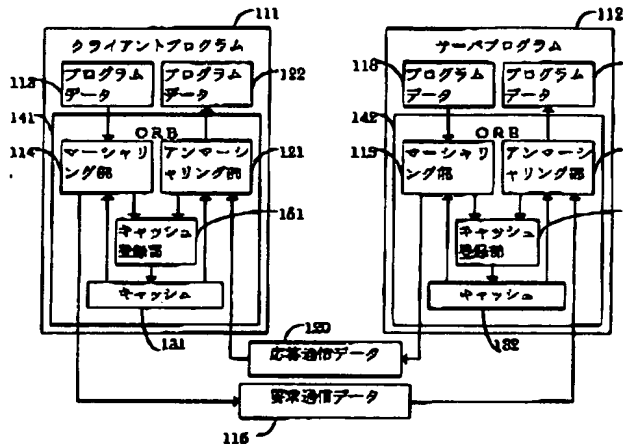
- 111 クライアントプログラム
- 112 サーバプログラム
- 114 クライアントプログラムのマーシャリング部
- 116 サーバプログラムのアンマーシャリング部
- 119 サーバプログラムのマーシャリング部
- クライアントプログラムのアンマーシャリング部
- 131 クライアントプログラムのキャッシュ部
- 132 サーバプログラムのキャッシュ部
- 141 クライアントプログラムのORB
- 142 クライアントプログラムのORB

(5)

特開平11-312151

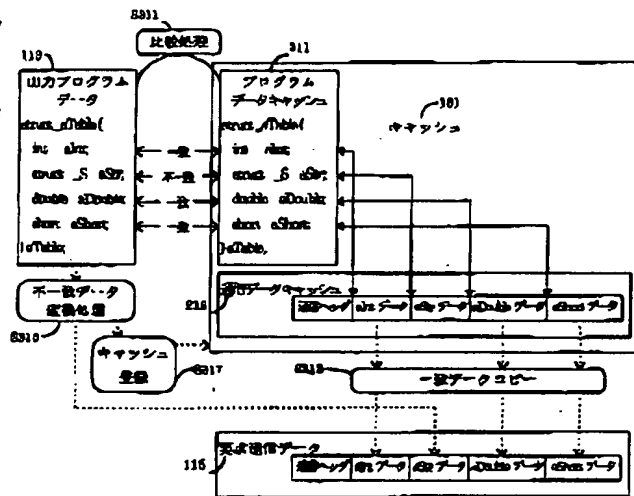
【図1】

【図1】



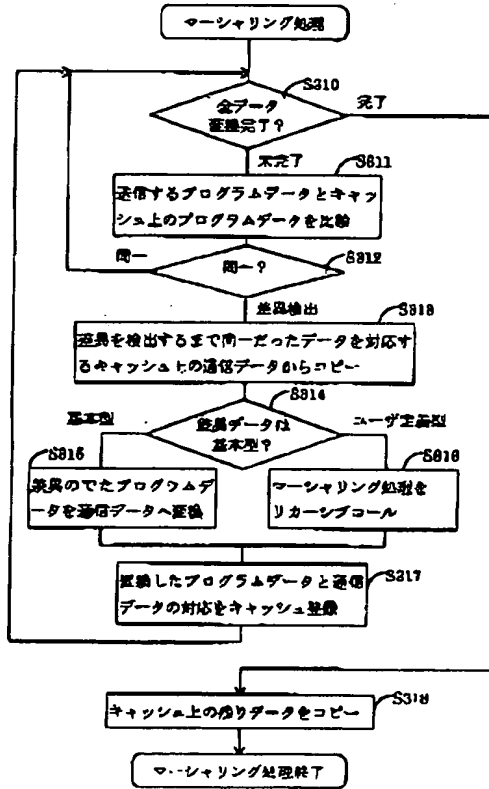
【図2】

【図2】



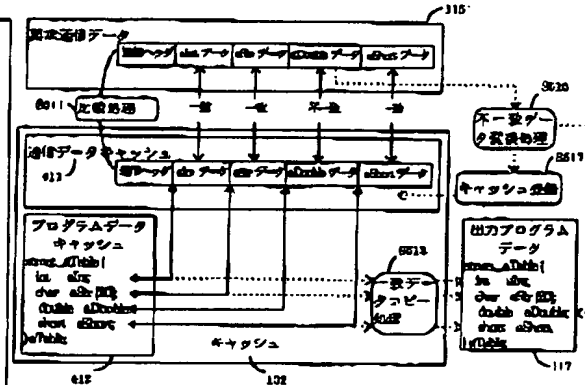
【図3】

【図3】



【図4】

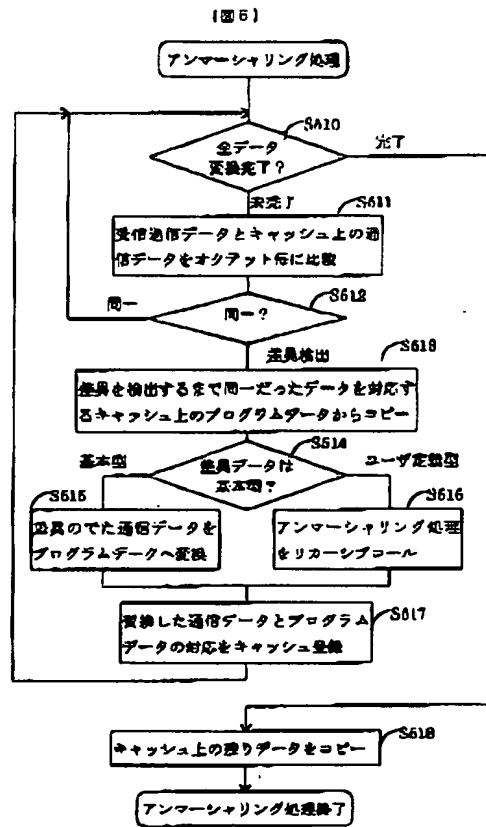
【図4】



(6)

特開平11-312151

【図5】





(7)

特開平 11-312151

【公報種別】特許法第17条の2の規定による補正の掲載

【部門区分】第6部門第3区分

【発行日】平成14年3月12日(2002. 3. 12)

【公開番号】特開平 11-312151

【公開日】平成11年11月9日(1999. 11. 9)

【年通号数】公開特許公報 11-3122

【出願番号】特願平 10-118121

【国際特許分類第7版】

G06F 15/16 430

370

9/46 360

13/00 351

【FI】

G06F 15/16 430 B

370 N

9/46 360 B

360 F

13/00 351 F

【手続補正書】

【提出日】平成13年9月19日(2001. 9. 19)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】分散コンピューティング環境において特定の計算機上の特定のプログラム言語依存のデータ形式を特定の計算機に依存しないストリーム化されたデータに変換する装置において、プログラム言語依存のデータ形式とストリーム化されたデータの対応を記憶するキャッシュ部と、プログラム言語依存のデータ形式の全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しない場合は、キャッシュを使用せずにプログラム言語依存のデータ形式をストリーム化されたデータに変換するマーシャリング部と、プログラム言語依存のデータ形式をストリーム化されたデータに変換、または、ストリーム化されたデータをプログラム言語依存のデータ形式に変換時に、その結果をキャッシュに登録するキャッシュ登録部を具備することを特徴とする装置。

【請求項2】分散コンピューティング環境において特定の計算機に依存しないストリーム化されたデータを特定の計算機上の特定のプログラム言語依存のデータ形式に

変換する装置において、

ストリーム化されたデータとプログラム言語依存のデータ形式の対応を記憶するキャッシュ部と、

ストリーム化されたデータの全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しなかった場合は、キャッシュを使用せずにストリーム化されたデータをプログラム言語依存のデータ形式に変換するアンマーシャリング部と、

ストリーム化されたデータをプログラム言語依存のデータ形式に変換、または、プログラム言語依存のデータ形式をストリーム化されたデータに変換時に、その結果をキャッシュに登録するキャッシュ登録部を具備することを特徴とする装置。

【請求項3】分散コンピューティング環境において特定の計算機上の特定のプログラム言語依存のデータ形式を特定の計算機に依存しないストリーム化されたデータに変換する方法において、

プログラム言語依存のデータ形式の全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッシュ上に存在しない場合は、キャッシュを使用せずにプログラム言語依存のデータ形式をストリーム化されたデータに変換し、

プログラム言語依存のデータ形式をストリーム化されたデータに変換、または、ストリーム化されたデータをプログラム言語依存のデータ形式に変換時に、その結果をキャッシュに登録することを特徴とする変換方法。

(8)

特開平11-312151

【請求項4】分散コンピューティング環境において特定の計算機に依存しないストリーム化されたデータを特定の計算機上の特定のプログラム言語依存のデータ形式に変換する変換方法において、

ストリーム化されたデータの全体又は一部がキャッシュに記憶されているかを検索しキャッシュ上に存在した場合はキャッシュ上のデータを用いて変換を行い、キャッ

シュ上に存在しなかった場合は、キャッシュを使用せずにストリーム化されたデータをプログラム言語依存のデータ形式に変換し、

ストリーム化されたデータをプログラム言語依存のデータ形式に変換、または、プログラム言語依存のデータ形式をストリーム化されたデータに変換時に、その結果をキャッシュに登録することを特徴とする変換方法。